

## Image Demosaicing and Enhancement System

## **Field of the Invention**

The present invention relates to digital cameras, and more particularly, to an improved method for converting data from a camera sensor to a color image.

5

## **Background of the Invention**

A digital color image usually consists of an array of pixel values representing the intensity of the image at each point on a regular grid. Typically, three colors are used to generate the image. At each point on the grid the intensity of each of these colors is specified, thereby specifying both the intensity and color of the image at that grid point.

10

Conventional color photography records the relevant image data by utilizing three overlapping color sensing layers having sensitivities in different regions of the spectrum (usually red, green, and blue). Digital cameras, in contrast, typically utilize one array of sensors in a single “layer”.

15

When only one sensor array is used to detect color images, only one color may be detected at any given sensor location. As a result, these sensors do not produce a color image in the traditional sense, but rather a collection of individual color samples, which depend upon the assignment of color filters to individual sensors. This assignment is referred to as the color filter array (CFA) or the color mosaic pattern. To produce a true color image, with a full set of color samples (usually red, green and blue) at each sampling location, a substantial amount of computation is required to estimate the missing information, since only a single color was originally sensed at each location in the array. This operation is typically referred to as “demosaicing”.

20

25

To generate the missing information, information from neighboring pixels in the image sensor must be used. A number of algorithms have been put forward in an attempt to provide the missing information while minimizing artifacts resulting from the estimation process.

5 The simplest algorithms interpolate the sensor data from like color sensors to provide the missing information. These algorithms treat the red sensors as being independent from the green sensors, and so on. To provide a red value at a given location, the values measured by the red sensors in the region of that location are interpolated. This approach requires that the image be low-pass filtered. Such filtering reduces the image resolution below the pixel resolution of the underlying sensor array. This lost resolution cannot be recovered.

To avoid this loss in resolution, less aggressive optical low-pass filtering is used in some higher-end cameras. However, in such systems, the color sensors may no longer be treated as independent. For example, Wober, et al. (U.S. Patent 5,475,769) describe a method for  
15 generating the missing color information by computing a weighted average of the pixel values in the neighborhood of the pixel whose missing color information is being computed. This method weights values from all of the color sensors, not just the color being reconstructed. However, even this approach leaves much to be desired since it utilizes one set of weights for all images.

20

A single pixel array may be viewed as consisting of a number of separate planes of pixels in which each plane has sensors for the same color. Since the pixels do not overlay, the sensors in the various planes are at different locations. Systems that take weighted averages across more than one plane make use of the statistical dependencies between these sample locations.

25 In effect, the blurring of an image by the camera optics allows an image edge that falls on one

00472202200

color plane precisely on the sensors of that plane to also be seen in the other color planes because the image is spread by blurring onto the sensors in the other color plane. Since the statistical dependencies between the various color planes depend on the amount of blur introduced by the camera optics, an optimal algorithm must take into account the physical camera settings. Accordingly, a single set of weight functions will not provide an optimal estimation of the missing information.

The statistical dependencies also depend on the source of illumination. Different illumination sources have different spectra. The pixel filters have broad pass-bands centered at the red, green, and blue wavelengths. In the absence of any image blurring, the response of any given pixel is determined by its color filter, the reflectivity of the corresponding point in the scene being photographed, and the light spectrum incident on that point from the illumination source. The blurring provided by the camera optics mixes the light between the pixels. Hence, the statistical dependencies, in general, depend both on the illumination source and the camera optics. Prior art methods for converting the pixel array data to a fully sampled color digital image do not take the illumination source into account.

Broadly, it is the object of the present invention to provide an improved image processing method for converting data from a pixel array having non-overlapping sensors to a fully sampled digital image.

It is a further object of the present invention to provide a conversion method that corrects for the camera's optical system.

It is a still further object of the present invention to provide a conversion method that corrects for the source of illumination.

These and other objects of the present invention will become apparent to those skilled in the art from the following detailed description of the invention and the accompanying drawings.

a

### Summary of the Invention

The present invention is a method for operating a data processing system to generate a second image from a first image. The first image includes a two dimensional array of pixel values, each pixel value corresponding to the light intensity in one of a plurality of spectral bands at a location in the first image. The method utilizes a linear transformation of a vector derived from super input pixels to obtain a vector that includes at least one super output pixel. The super input pixels are defined by separating the pixels of the first image into a plurality of input image planes. Each input image plane has an identical number of pixels within a normalized horizontal and vertical sampling interval as the other input image planes. All pixels in a given input image plane correspond to the same spectral band as the other pixels in that input image plane. Each super input pixel is a vector of dimension  $P$ , where  $P$  is the number of the input image planes, each component of that vector being an input pixel from a corresponding input image plane. Similarly, a set of output image planes is defined, each pixel in a given output image plane representing the intensity of the second image in one of a plurality of spectral bands at a corresponding point in the second image. Each super output pixel is a vector of dimension  $Q$ , where  $Q$  is the number of the output image planes, each component of that vector being a pixel from a corresponding output image plane. In the preferred embodiment of the present invention, the linear transformation depends on the

properties of the optical system and the illumination source used to generate the first image. The linear transformation can also be varied to take into account the contents of the scene captured in the first image and the desired output format of the second image.

### **Brief Description of the Drawings**

Figure 1 illustrates the separation of an image taken with an image sensor having a repeating 2x2 pattern into image planes according to the present invention.

Figure 2 illustrates a portion of a sensor array and the input pixels in the sensor array which contribute to a particular intermediate input vector.

Figure 3 illustrates a portion of an output RGB (red, green, blue) image and the pixels in the RGB output image that correspond to the intermediate output vector shown in Figure 2.

### **Detailed Description of the Invention**

The method of the present invention may be applied to any color-sampling device that acquires its samples from a sensor array that can be decomposed into a plurality of image planes, which satisfy two conditions. First, each image plane must have an identical number of samples within a normalized horizontal and vertical sampling interval; however, the various image planes may be arbitrarily displaced relative to one another. Second, all samples in a given image plane must have identical color properties; however, multiple image planes can have the same color properties.

suba1  
These conditions are satisfied by any image sensor having a sensor pattern that is constructed by repeating a kernel of sensing elements. For example, one common image sensor array is based on the Bayer pattern which generated by repeating a 2x2 sensor array kernel having two green sensors, one red sensor, and one blue sensor. This pattern is shown in Figure 1 at 10. The kernel is shown at 12. Such an image sensor may be viewed as having 4 planes shown at 14-17, two green planes 14 and 17, one red plane 16, and one blue plane 15. The sampling interval is the area originally occupied by the kernel. Each of the planes is offset with respect to the other planes. It can be shown that any regular sampling lattice can be decomposed into a set of image planes satisfying the above conditions.

To simplify the following discussion, vector notation will be utilized. Vectors and matrices will be shown in bold print to distinguish them from scalar quantities. The measured intensity values in each image plane will be denoted by  $x_p[n_1, n_2]$ . Here,  $n_1$  and  $n_2$  are indices which denote the position of the pixel in the  $p^{\text{th}}$  image plane and  $x_p$  is the intensity value measured for that pixel. The quantity  $[n_1, n_2]$  is a two-dimensional integer valued vector which will be denoted by  $\mathbf{n}$ . The entire set of image planes can then be represented as a set of vectors  $\mathbf{x}[\mathbf{n}]$  where

$$\mathbf{x}[\mathbf{n}] = \begin{bmatrix} \mathbf{x}_1[n_1, n_2] \\ \mathbf{x}_2[n_1, n_2] \\ \vdots \\ \mathbf{x}_p[n_1, n_2] \end{bmatrix} \quad (1)$$

The output image can likewise be represented as a set of vectors defined on a different set of image planes. Typically, the goal of the demosaicing algorithm is to generate a set of regularly spaced pixels in three color planes (red, green, and blue). Denote the intensity in

the  $i^{\text{th}}$  color plane by  $y_i[n_1, n_2]$ . Then the output pixels can also be represented by a set of vectors.

$$\mathbf{y}[\mathbf{m}] = \begin{bmatrix} y_1[m_1, m_2] \\ y_2[m_1, m_2] \\ \vdots \\ y_Q[m_1, m_2] \end{bmatrix} \quad (2)$$

- 5 In the demosaicing case,  $Q$  is typically 3; however, different  $Q$  values may be utilized. For example, an image that is to be printed on a color printer utilizing 4 dyes could be generated directly by the method of the present invention utilizing a representation in which  $Q=4$ .

In general, the output image will have a spatial resolution that is different from the input  
 10 image. The input image may be viewed as consisting of a set of “super pixels”,  $\mathbf{x}[\mathbf{n}]$ . Likewise, the output image is a set of pixels  $\mathbf{y}[\mathbf{m}]$ . The number of output pixels in the vertical and horizontal directions corresponding to each input pixel will be denoted by  $\lambda_1$  and  $\lambda_2$ , respectively. In the case of the Bayer pattern discussed above, the demosaicing task is usually understood as having  $\lambda_1 = \lambda_2 = 2$ . That is, one attempts to construct one output (RGB)  
 15 pixel for each physical sensor in the input array.

In the method of the present invention, the output pixels are related to the input pixels by a linear operator that operates on vectors derived from  $\mathbf{x}[\mathbf{n}]$  and  $\mathbf{y}[\mathbf{m}]$ . These intermediate vectors take into account the difference in resolution and the fact that each output pixel  
 20 depends on more than one input super pixel. The intermediate vector corresponding to  $\mathbf{y}[\mathbf{m}]$  will be denoted by  $\zeta[\mathbf{n}]$  and has the same sampling density as  $\mathbf{x}[\mathbf{n}]$ :

$$\zeta[\mathbf{n}] = \begin{bmatrix} y[\Lambda(\mathbf{n} + \delta_1)] \\ y[\Lambda(\mathbf{n} + \delta_2)] \\ \vdots \\ y[\Lambda(\mathbf{n} + \delta_{\lambda_1, \lambda_2})] \end{bmatrix} \quad (3)$$

Here, the matrix  $\Lambda$  is defined by

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \quad (4)$$

In the case of the Bayer pattern,  $\delta_1=[0,0]$ ,  $\delta_2=[1/2,0]$ ,  $\delta_3=[0,1/2]$ , and  $\delta_4=[1/2,1/2]$ . The vectors  $\zeta[\mathbf{n}]$  will be referred to as the output polyphase components in the following discussion.

In the method of the present invention, it is assumed that each polyphase output vector depends on a finite number of input super pixels. In general, the input super pixels that contribute to a particular polyphase output vector  $\zeta[\mathbf{n}]$  will be located in a neighborhood around  $[\mathbf{n}]$ . As will be explained in more detail below, the precise pixels will depend on the nature of the camera and imaging optics. The input super pixels that contribute to the polyphase output vector at  $\mathbf{n}$  may be identified by a set of displacement vectors  $\mathbf{k}_1, \mathbf{k}_2, \dots, \mathbf{k}_K$ . That is,  $\zeta[\mathbf{n}]$  depends on  $\mathbf{x}[\mathbf{n}+\mathbf{k}_1], \mathbf{x}[\mathbf{n}+\mathbf{k}_2], \dots, \mathbf{x}[\mathbf{n}+\mathbf{k}_K]$ . In the method of the present invention,  $\zeta[\mathbf{n}]$  is assumed to be linearly dependent on the input super pixels. In the preferred embodiment of the present invention, the set of displacement vectors  $\mathbf{k}_1, \mathbf{k}_2, \dots, \mathbf{k}_K$  is independent of  $\mathbf{n}$ , and is arranged in a  $k_1 \times k_2$  rectangular grid

The linear relationship can be most easily defined in terms of a vector  $\xi[\mathbf{n}]$  which includes all of the super pixels on which the output polyphase vector  $\zeta[\mathbf{n}]$  depends, i.e.,



$$\xi[\mathbf{n}] = \begin{bmatrix} \mathbf{x}[\mathbf{n} + \mathbf{k}_1] \\ \vdots \\ \mathbf{x}[\mathbf{n} + \mathbf{k}_K] \end{bmatrix} \quad (5)$$

In terms of this vector, the relationship between  $\zeta[\mathbf{n}]$  and  $\xi[\mathbf{n}]$  may be written as matrix multiplication:

$$\zeta[\mathbf{n}] = \mathbf{T}\xi[\mathbf{n}] \quad (6)$$

5

where  $\mathbf{T}$  is a  $(Q\lambda_1\lambda_2) \times (PK_1K_2)$  matrix.

Refer now to Figures 2 and 3 which illustrate the relationships between the output pixels  $Y_j[N,M]$ , the input pixels  $x_i[N,M]$  and the two intermediate vectors defined above for the Bayer sensor pattern. Figure 2 illustrates a portion of a sensor array and the input pixels in the sensor array which contribute to  $\xi[N,M]$ . Figure 3 illustrates a portion of an output RGB image and the pixels in the RGB output image that correspond to  $\zeta[N,M]$  and which are computed from the pixels shown in Figure 2 by the matrix multiplication operation shown in Eq. (6).

15

The matrix,  $\mathbf{T}$ , depends on a number of factors. Some of these are fixed for a particular imaging device and some depend on the particular manner in which the imaging device is being utilized. For example, the physical properties of the sensing array such as the spectral sensitivity of the pixels, the mosaic pattern, and the number of pixels typically do not vary from image to image. In contrast, the optical properties of the imaging device such as the lens settings on a camera (f number and zoom) may vary from image to image. In addition, the spectral properties of the illumination source may vary from image to image (daylight, flash, incandescent light, etc.).

20

00477722-102300

In addition, the statistics of the image being captured may be taken into account through  $T$ . For example, images or portions of images having a high content of vertical and horizontal edges can be processed with a different matrix than images lacking such features and thereby improve the output image quality.

In cameras that have a variable output format, the resolution of the final picture can be set using a different  $T$  matrix. Alternatively, a single  $T$  matrix may be utilized for all resolutions and then the desired output image determined by re-sampling the fixed resolution image.

Similarly, the number of output color planes may be altered by using different  $T$  matrices or by resampling a single color format to generate an alternate color representation. In general, properties that alter the dimension of the  $T$  matrix are preferably handled by using a fixed  $T$  matrix and then re-sampling the final image in low cost imaging devices.

If the number of different  $T$  matrices is relatively small, the coefficients of the  $T$  matrix can be determined by training the system on known images. For each possible  $T$  matrix, images of a number of known scenes are taken using the imaging device. The coefficients of the  $T$  matrix are then computed so as to minimize the difference between the image computed from the sensor input and the known scene images. Such optimization computations are well known to those skilled in the art, and hence, will not be discussed in detail here.

If the variation in some parameter such as f-number is relatively smooth, the  $T$  matrices need only be computed for a discrete number of values of the parameter. The correct  $T$  matrix for the non-computed variable parameter values can then be computed by interpolation of the computed  $T$  matrices.

## Model-Based Computation of $\mathbf{T}$

As noted previously, in some circumstances it may be possible to compute appropriate matrices,  $\mathbf{T}$ , from training images. Unfortunately, this approach is limited to applications in which the number of different imaging conditions and hence the number of different  $\mathbf{T}$  matrices which could be required is relatively small. The purpose of the material presented here is to describe a method for directly computing  $\mathbf{T}$  for an arbitrary imaging device (i.e. arbitrary color sensitivities, sensor locations and optical characteristics) and under arbitrary illumination, subject to a particular statistical model for the underlying image, which has been found to give particularly good reconstructed image quality. As will be seen, the statistical image model is governed by only a few parameters. In more advanced applications, these parameters may be adjusted, either locally, or globally, to match statistical properties of the image, such as edge orientation, which can be estimated by various methods.

## Image Formation Model

This section describes the parameters of the image formation process which maps the original scene into the source image super-pixels,  $\mathbf{x}[\mathbf{n}]$ . The image formation model depends upon deterministic quantities which can, at least in theory, be measured. These quantities are

- The scene illuminant spectral power density,  $l(\lambda)$ .
- The color spectral response functions,  $r_p(\lambda)$ , for each input image plane,  $p$ .
- The Point Spread Function,  $h_p(\lambda, \mathbf{s})$ , associated with the combined effects of the optical transfer function and sensor integration behaviour for input image plane  $p$ . Here  $\mathbf{s} = [s_1, s_2]$  is the spatially continuous argument of the Point Spread Function (PSF), at each wavelength,  $\lambda$ . Note that the PSF's implicitly include

the effect of relative displacements between the different input image planes. In the following, the PSF is referenced only through its Fourier Transform,  $\hat{h}_p(\lambda\omega)$ , where the spatial frequency vector,  $\omega = [\omega_1, \omega_2]$ , is normalized so that  $\omega_1 = \omega_2 = \pi$  at the Nyquist frequency of the input super-pixels. Thus, the

5 Nyquist frequency of the original sensor array corresponds to  $\omega = [\lambda_1\pi, \lambda_2\pi]$ .

Rather than modeling the image formation process directly in terms of the desired output image super-pixels, it is helpful to choose an intermediate representation in terms of surface spectral-reflectance, since this is well known to be better behaved from a statistical perspective than the scene radiance itself and hence more amenable to the statistical modelling described in the next section. Specifically it is helpful to assume that the spectral reflectance of the original scene can be perfectly represented as a linear combination of a limited number of fixed basis functions,  $b_1(\lambda), b_2(\lambda), \dots, b_S(\lambda)$ , where  $S$  is usually chosen to be three or four, but may be larger if desired. The actual output vectors,  $\mathbf{y}[\mathbf{m}]$ , may be expressed in terms of the intermediate spectral reflectance vectors,  $\mathbf{z}[\mathbf{m}]$  as

10

15

$$\mathbf{y}[\mathbf{m}] = \mathbf{T}_{\text{out}} \cdot \mathbf{z}[\mathbf{m}]$$

$$= \begin{pmatrix} \int_0^\infty d_1(\lambda)l(\lambda)b_1(\lambda)d\lambda & \int_0^\infty d_1(\lambda)l(\lambda)b_2(\lambda)d\lambda & \cdots & \int_0^\infty d_1(\lambda)l(\lambda)b_S(\lambda)d\lambda \\ \int_0^\infty d_2(\lambda)l(\lambda)b_1(\lambda)d\lambda & \int_0^\infty d_2(\lambda)l(\lambda)b_2(\lambda)d\lambda & \cdots & \int_0^\infty d_2(\lambda)l(\lambda)b_S(\lambda)d\lambda \\ \vdots & \vdots & \ddots & \vdots \\ \int_0^\infty d_Q(\lambda)l(\lambda)b_1(\lambda)d\lambda & \int_0^\infty d_Q(\lambda)l(\lambda)b_2(\lambda)d\lambda & \cdots & \int_0^\infty d_Q(\lambda)l(\lambda)b_S(\lambda)d\lambda \end{pmatrix} \cdot \mathbf{z}[\mathbf{m}]$$

20

where  $d_q(\lambda)$  is the spectral response of the  $q$ 'th display spectral response function. For example, if the objective is to recover an XYZ image, then  $Q$  should be set to 3 and  $d_1(\lambda)$  through  $d_3(\lambda)$  should be set to the standard 1931 CIE tri-stimulus functions. As another example, if the objective is to recover an image with the same color characteristics as the different color filters on the physical sensor array, then  $Q$  should be set to the number of unique input response functions,  $r_p(\lambda)$ , and there should

25

be a one-to-one correspondence between these unique  $r_p(\lambda)$  and the  $d_q(\lambda)$ . In this framework, the chief goal is to compute the  $(S\lambda_1\lambda_2) \times (PK_1K_2)$  reconstruction matrix,  $\mathbf{T}_{ref}$  which maps the neighbourhood of input super-pixels,  $\xi[\mathbf{n}]$ , to the corresponding spectral-reflectance super-pixel,

$$\zeta'[\mathbf{n}] = \begin{pmatrix} \mathbf{z}[\mathbf{\Lambda}(\mathbf{n} + \delta_1)] \\ \mathbf{z}[\mathbf{\Lambda}(\mathbf{n} + \delta_2)] \\ \vdots \\ \mathbf{z}[\mathbf{\Lambda}(\mathbf{n} + \delta_{\lambda_1\lambda_2})] \end{pmatrix}$$

The final  $(Q\lambda_1\lambda_2) \times (PK_1K_2)$  reconstruction matrix is then formed by simple matrix multiplication:

$$\mathbf{T} = \begin{pmatrix} \mathbf{T}_{out} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}_{out} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & & \ddots & & \vdots & \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{T}_{out} \end{pmatrix} \cdot \mathbf{T}_{ref} \quad (7)$$

The linear image formation model may now be expressed compactly as

$$\hat{\mathbf{x}}(\omega) = \hat{\nu}(\omega) + \sum_{\omega_a \in \Omega_a(\omega)} \mathbf{H}(\omega_a) \hat{\mathbf{z}}(\mathbf{\Lambda}^{-1}\omega_a)$$

where

- $\hat{\mathbf{x}}(\omega)$  is the Discrete Space Fourier Transform of the input image,  $\mathbf{x}[\mathbf{n}]$ ;
- $\hat{\nu}(\omega)$  is the Discrete Space Fourier Transform of the sampling noise vector sequence;
- $\hat{\mathbf{z}}(\omega)$  is the Discrete Space Fourier Transform of the spectral reflectance vector,  $\mathbf{z}[\mathbf{m}]$ ;

- $\Omega_a(\omega)$ , is the set containing all  $\lambda_1 \lambda_2$  aliasing frequencies associated with the sampling of the high resolution output image on grid  $[\mathbf{m}]$  onto the input super-pixel grid  $[\mathbf{n}]$ , for each  $\omega \in [-\pi, \pi]^2$ ; and
- $\mathbf{H}(\omega)$  is the  $P \times S$  image formation matrix,

$$\mathbf{H}(\omega) = \begin{pmatrix} \int_0^\infty r_1(\lambda) l(\lambda) b_1(\lambda) \hat{h}_1(\lambda, \omega) d\lambda & \cdots & \int_0^\infty r_1(\lambda) l(\lambda) b_S(\lambda) \hat{h}_1(\lambda, \omega) d\lambda \\ \int_0^\infty r_2(\lambda) l(\lambda) b_1(\lambda) \hat{h}_2(\lambda, \omega) d\lambda & \cdots & \int_0^\infty r_2(\lambda) l(\lambda) b_S(\lambda) \hat{h}_2(\lambda, \omega) d\lambda \\ \vdots & \ddots & \vdots \\ \int_0^\infty r_P(\lambda) l(\lambda) b_1(\lambda) \hat{h}_P(\lambda, \omega) d\lambda & \cdots & \int_0^\infty r_P(\lambda) l(\lambda) b_S(\lambda) \hat{h}_P(\lambda, \omega) d\lambda \end{pmatrix}$$

## Statistical Model

In order to compute an appropriate solution to the image reconstruction problem, it is necessary to introduce a statistical model for the sampling noise,  $\hat{\nu}(\omega)$ , and the spectral reflectance,  $\hat{\mathbf{z}}(\omega)$ . In this discussion, Wide Sense Stationary Gaussian models are assumed, which are characterized entirely by the covariance matrices,

$$\mathbf{C}_\nu(\omega) = E [\hat{\nu}(\omega) \cdot \hat{\nu}(\omega)^*]$$

and

$$\mathbf{C}_z(\omega) = E [\hat{\mathbf{z}}(\omega) \cdot \hat{\mathbf{z}}(\omega)^*]$$

The noise covariance matrix will usually be a constant,  $\mathbf{C}_\nu(\omega) = \sigma^2 \mathbf{I}$ , for all  $\omega$ , corresponding to white noise, but other models may be used if appropriate.

The following parametric model is used for the reflectance covariance matrix,

$$\mathbf{C}_z(\omega) = \mathbf{C}_z^0 \cdot \|\Gamma \omega\|^{-\frac{\rho}{10}} \quad (8)$$

where  $\mathbf{C}_z^0$  is a constant  $S \times S$  covariance matrix,  $\rho$  is a frequency roll-off parameter, which is usually selected in the range 20 to 30 dB/decade, and  $\Gamma$  is a  $2 \times 2$  “shape matrix”. The terms in the above expression which follow the constant covariance

matrix,  $\mathbf{C}_z^0$ , describe a scalar envelope function whose contours are ellipses in the frequency domain. The orientation and aspect ratio of these elliptic contours may be explicitly controlled by means of the  $\mathbf{\Gamma}$  matrix. For a circular cross-section, the identity matrix,  $\mathbf{\Gamma} = \mathbf{I}$ , may be used.

The statistical model represented by  $\mathbf{C}_z$  plays an extremely important role in determining the quality of the final reconstructed images. The parametric model described above may be justified on a number of grounds; most notably, the model is scale-invariant, which means that on average the statistics of scenes should not depend upon how far the camera is located from the scene. This scale-invariance property is important because in practical imaging applications, information about the absolute scale of objects in the scene is rarely available. Also, there is significant empirical evidence for this scale invariance property in natural scenes, with a frequency roll-off factor,  $\rho$ , of about 20 dB/decade.

## Efficient Computation of $\mathbf{T}$

As mentioned above, the key objective is to compute the  $(S\lambda_1\lambda_2) \times (PK_1K_2)$  matrix,  $\mathbf{T}_{\text{ref}}$ , from which  $\mathbf{T}$  is easily recovered via equation (7). The ensuing discussion concerns the derivation of an optimal Linear Minimum Mean Squared Error (LMMSE) estimator,  $\mathbf{T}_{\text{ref}}$ , subject to the models described in the previous two sections. The formula for such an estimator is well-known. Specifically,

$$\mathbf{T}_{\text{ref}} = \mathbf{Z} \cdot \mathbf{X}^{-1} \quad (9)$$

where  $\mathbf{Z}$  is the  $(S\lambda_1\lambda_2) \times (PK_1K_2)$  cross-covariance matrix,

$$\mathbf{Z} = E [\zeta'[\mathbf{n}] \cdot \xi[\mathbf{n}]^t]$$

and  $\mathbf{X}$  is the  $(PK_1K_2) \times (PK_1K_2)$  dimensional auto-covariance matrix,

$$\mathbf{X} = E [\xi[\mathbf{n}] \cdot \xi[\mathbf{n}]^t]$$

In fact,  $\mathbf{X}$  has the following Toeplitz block-Toeplitz structure,

$$\mathbf{X} = \begin{pmatrix} \underline{\mathbf{X}}[0] & \underline{\mathbf{X}}[1] & \cdots & \underline{\mathbf{X}}[K_2 - 1] \\ \underline{\mathbf{X}}[-1] & \underline{\mathbf{X}}[0] & \cdots & \underline{\mathbf{X}}[K_2 - 2] \\ \vdots & & \ddots & \vdots \\ \underline{\mathbf{X}}[1 - K_2] & \underline{\mathbf{X}}[2 - K_2] & \cdots & \underline{\mathbf{X}}[0] \end{pmatrix}$$

where each block,  $\underline{\mathbf{X}}[l_2]$ , has the Toeplitz form

$$\underline{\mathbf{X}}[l_2] = \begin{pmatrix} \underline{\underline{\mathbf{X}}}[0, l_2] & \underline{\underline{\mathbf{X}}}[1, l_2] & \cdots & \underline{\underline{\mathbf{X}}}[K_1 - 1, l_2] \\ \underline{\underline{\mathbf{X}}}[-1, l_2] & \underline{\underline{\mathbf{X}}}[0, l_2] & \cdots & \underline{\underline{\mathbf{X}}}[K_1 - 2, l_2] \\ \vdots & & \ddots & \vdots \\ \underline{\underline{\mathbf{X}}}[1 - K_1, l_2] & \underline{\underline{\mathbf{X}}}[2 - K_1, l_2] & \cdots & \underline{\underline{\mathbf{X}}}[0, l_2] \end{pmatrix}$$

and each sub-block,  $\underline{\underline{\mathbf{X}}}[l_1, l_2]$ , is a  $P \times P$  source super-pixel covariance matrix, given by

$$\underline{\underline{\mathbf{X}}}[\mathbf{l}] = E [\mathbf{x}[\mathbf{n}] \cdot \mathbf{x}[\mathbf{n} + \mathbf{l}]^t]$$

The  $(S\lambda_1\lambda_2) \times (PK_1K_2)$  matrix,  $\mathbf{Z}$ , also has a doubly-nested block structure. Specifically,

$$\mathbf{Z} = \left( \underline{\underline{\mathbf{Z}}} \left[ -\left\lfloor \frac{K_2}{2} \right\rfloor \right] \quad \underline{\underline{\mathbf{Z}}} \left[ 1 - \left\lfloor \frac{K_2}{2} \right\rfloor \right] \quad \cdots \quad \underline{\underline{\mathbf{Z}}} \left[ \left\lfloor \frac{K_2 - 1}{2} \right\rfloor \right] \right)$$

where

$$\underline{\underline{\mathbf{Z}}}[l_2] = \left( \underline{\underline{\underline{\mathbf{Z}}}} \left[ -\left\lfloor \frac{K_1}{2} \right\rfloor, l_2 \right] \quad \underline{\underline{\underline{\mathbf{Z}}}} \left[ 1 - \left\lfloor \frac{K_1}{2} \right\rfloor, l_2 \right] \quad \cdots \quad \underline{\underline{\underline{\mathbf{Z}}}} \left[ \left\lfloor \frac{K_1 - 1}{2} \right\rfloor, l_2 \right] \right)$$

and the  $(S\lambda_1\lambda_2) \times P$  sub-block matrices,  $\underline{\underline{\mathbf{Z}}}[\mathbf{l}]$ , are given by

$$\underline{\underline{\mathbf{Z}}}[\mathbf{l}] = E [\zeta'[\mathbf{n}] \cdot \xi[\mathbf{n} + \mathbf{l}]^t].$$

In order to compute  $\mathbf{T}_{\text{ref}}$ , then, it is sufficient to compute the matrices  $\underline{\underline{\mathbf{X}}}[\mathbf{l}]$  for  $[\mathbf{l}] = [l_1, l_2]$  in the range  $-K_i < l_i < K_i$  and the matrices  $\underline{\underline{\mathbf{Z}}}[\mathbf{l}]$ , for  $[\mathbf{l}] = [l_1, l_2]$  in the range  $-\left\lfloor \frac{K_i}{2} \right\rfloor \leq l_i \leq \left\lfloor \frac{K_i - 1}{2} \right\rfloor$ , after which the contents of  $\mathbf{X}$  and  $\mathbf{Z}$  may be filled in



and used to evaluate equation (9). The key to efficient computation of  $\mathbf{T}_{\text{ref}}$  lies in efficiently computing the matrices,  $\underline{\mathbf{X}}[\mathbf{l}]$  and  $\underline{\mathbf{Z}}[\mathbf{l}]$ .

It turns out that these matrices may be efficiently computed by exploiting Parseval's Relationship. Specifically,

$$\underline{\mathbf{X}}[-\mathbf{l}] = \frac{1}{(2\pi)^2} \int_{-\pi}^{\pi} d\omega_1 \int_{-\pi}^{\pi} d\omega_2 e^{j\omega^t \mathbf{l}} \mathbf{C}_{\mathbf{x}}(\omega) \quad (10)$$

and

$$\underline{\mathbf{Z}}[-\mathbf{l}] = \frac{1}{(2\pi)^2} \int_{-\pi}^{\pi} d\omega_1 \int_{-\pi}^{\pi} d\omega_2 e^{j\omega^t \mathbf{l}} \mathbf{C}_{\zeta' \mathbf{x}}(\omega) \quad (11)$$

where the frequency auto- and cross-covariance matrices,  $\mathbf{C}_{\mathbf{x}}(\omega)$  and  $\mathbf{C}_{\zeta' \mathbf{x}}(\omega)$  are found from

$$\mathbf{C}_{\mathbf{x}}(\omega) = \mathbf{C}_{\nu}(\omega) + \sum_{\omega_a \in \Omega_a(\omega)} \left( \mathbf{H}(\omega_a) \mathbf{C}_{\mathbf{z}}(\omega_a) \mathbf{H}(\omega_a)^* \right) \quad (12)$$

and

$$\mathbf{C}_{\zeta' \mathbf{x}}(\omega) = \sum_{\omega_a \in \Omega_a(\omega)} \left( \mathbf{\Phi}(\omega_a) \mathbf{C}_{\mathbf{z}}(\omega_a) \mathbf{H}(\omega_a)^* \right) \quad (13)$$

Here,  $\mathbf{\Phi}(\omega)$  is the  $(S\lambda_1\lambda_2) \times S$  matrix of phase shifts,

$$\mathbf{\Phi}(\omega) = \begin{pmatrix} e^{j\omega^t \delta_1} \mathbf{I}(S) \\ e^{j\omega^t \delta_2} \mathbf{I}(S) \\ \vdots \\ e^{j\omega^t \delta_{\lambda_1 \lambda_2}} \mathbf{I}(S) \end{pmatrix}$$

corresponding to the relative displacements of each of the output polyphase components.

In order to compute the matrices,  $\underline{\mathbf{X}}[\mathbf{l}]$  and  $\underline{\mathbf{Z}}[\mathbf{l}]$ ,  $\mathbf{C}_{\zeta' \mathbf{x}}(\omega)$  and  $\mathbf{C}_{\mathbf{x}}(\omega)$  are evaluated at a finite number of frequencies,  $\omega \in [-\pi, \pi]^2$  and then the Inverse Fourier Transform (IFT) integrals of equations (10) and (11) are approximated numerically. There are various approaches to determining the best set of frequency points at which to evaluate  $\underline{\mathbf{X}}[\mathbf{l}]$  and  $\underline{\mathbf{Z}}[\mathbf{l}]$  and interpolating between these points during the numerical integration procedure, but these are beyond the scope of this brief discussion.

Various modifications to the present invention will become apparent to those skilled in the art from the foregoing description and accompanying drawings. Accordingly, the present invention is to be limited solely by the scope of the following claims.